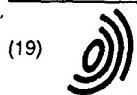


AJ



Europäisches Patentamt

European Patent Office

Office européen des brevets



(11) EP 0 810 520 A1

(12)

EUROPEAN PATENT APPLICATION

(43) Date of publication:
03.12.1997 Bulletin 1997/49

(51) Int. Cl.⁶: G06F 9/44, G06F 17/30

(21) Application number: 97201341.1

(22) Date of filing: 03.05.1997

(84) Designated Contracting States:
DE FR GB NL SE

(72) Inventor: Smith, Jody K.
Colorado Springs, Colorado 80903 (US)

(30) Priority: 28.05.1996 US 654646

(74) Representative:
Hanna, Peter William Derek et al
Tomkins & Co., --
5 Dartmouth Road
Dublin 6 (IE)

(71) Applicant: SUN MICROSYSTEMS INC.
Mountain View, California 94043-1100 (US)

(54) Dynamic linkable labels in a network browser page

(57) In a computer network (46) where remote user stations (35) retrieve information from other sites in the network, a method and apparatus are disclosed for creating and displaying dynamic link labels in a browser program (50) operating on a remote user station. The link labels are created in an application program (45) which can be run within the browser, and the link labels are designed to operate, at a minimum, in a similar manner as HTML (Hyper Text Markup Language) hyper links. The link labels can also dynamically change in response to user input into the browser. For instance, the URL (Uniform Resource Locator) address or the text or appearance of the link label can change. Also, parameters based on user input can be formed by the application and used to form or alter other link labels.

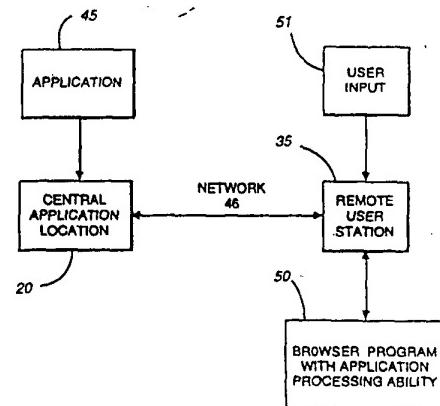


Fig. 2

EP 0 810 520 A1

state of mouse buttons or other inputs. A link label modification module modifies the link label in response to the user input by associating the link label with a second network address. The actual text or visual appearance of the link label can also be changed. Finally, a parameters formation module can form passable parameters based on the user input into the browser, and these parameters can be passed to other application programs downloaded to the remote user station.

The great advantage or utility of the invention is that the labels within a web page can be dynamically changed so that the destination URL address can be dynamically changed in response to user input within the web page, or from receipt of dynamic parameters passed by remote computing stations in the network.

Another utility of the invention is that the textual content of the link within the web page can also be dynamically changed in response to user input into the web page or dynamic parameters passed by remote computing stations.

Another utility of the invention is that the link label can operate with characteristics resembling a hyper link, so that user's familiarity with conventional links is maintained with the use of link labels of the present invention.

Another utility of the invention is that a programmer can design a web page with a plurality of link labels to be responsive to the user inputs into the web browser page running on the remote computing station.

Another utility of the present invention is that a link in a page of a browser can be dynamically altered without necessitating any additional network transmissions, thereby improving the system's performance by reducing network access times and avoiding network latency.

The foregoing and other features, utilities, and advantages of the invention will be apparent from the following more particular description of a preferred embodiment of the invention as illustrated in the accompanying drawings.

BRIEF DESCRIPTION OF DRAWINGS

Fig. 1 illustrates a computing system acting as a client-server in a communications network, such as the Internet, to perform the logical operations of the invention.

Fig. 2 is a block diagram of the preferred embodiment of the present invention.

Figs. 3A through 3C illustrate an example of a web page utilizing the link labels of the present invention.

Fig. 4 illustrates the logical operations of the preferred embodiment of the invention to create and display a link label in a browser page.

Fig. 5 illustrates the logical operations of creating a window in a browser having a link label.

Fig. 6 illustrates the logical operations of processing user input to alter a link label of the present invention displayed in a browser page.

Figs. 7A through 7B illustrate the logical operations

of detecting the user's input into a web browser page to utilize a link label of the present invention.

Fig. 8 illustrates an alternative embodiment of the present invention for passing data parameters between application programs for dynamic processing therein.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

10 The embodiments of the invention described herein are implemented as logical operations in a computing system. The logical operations of the present invention are implemented (1) as a sequence of computer implemented steps running on the computing system and (2) as interconnected machine modules within the computing system. The implementation is a matter of choice dependent on the performance requirements of the computing system implementing the invention. Accordingly, the logical operations making up the embodiments of the invention described herein are referred to variously as operations, steps, or modules.

15 The operating environment in which the present invention is used encompasses general distributed computing systems wherein general purpose computers, work stations, or personal computers are connected via communication links of various types. In a client server arrangement, programs and data, many in the form of objects, are made available by various members of the system.

20 In accordance with the invention, users at remote terminals in a network communicate through the network to a computer server or a web site and are able to download data from the server or web site to the user's work station.

25 As this embodiment of the invention is described herein, a web browser program on a client station for browsing a network, such as the Internet, will be referred to as the browser, while the server work station with which the browser station is communicating during a download will be referred to as the server or the central application location.

30 Referring to Fig. 1, data processing system 20, acting as a server through an application program 45, places a plurality of web pages 40 for access by remote client stations 35 over network 46. These web pages 40 are originated by the application program 45. Web pages 40 can contain data including text, graphics, audio files, video files, and other forms of data. Included in web pages 40 are links which have textual labels and URL destination addresses associated therewith.

35 In Fig. 1, processor 21 of server 20 includes an input/output section 22, a central processing unit 23 (CPU), and a memory section 24. The input/output section 22 is connected to a keyboard 25, a display or monitor 26, and a disk storage unit 29 as well as an optical disk drive unit 27 for reading a CD ROM 28, and a floppy disk drive unit 30 for reading a diskette 31. Application program 45 may be loaded into the server 20 from either the CD ROM 28 or diskette 31. The computer pro-

45 from the central application location 20 over network 46. Operation 60 can occur in response to user input, such as the user selecting through browser program 50 (Fig. 2) a web site having an application program available for download.

In the preferred embodiment of the present invention, downloading a desired application over the network appears, from the user's perspective, identical to downloading information generally from a web site over the Internet. The fact that the user is downloading an application designed to run within the browser of the user's remote station may be transparent to the user, as this operation can be made to appear identical to a download of a traditional HTML web page.

Operation 62 runs the downloaded application on the browser of the remote user station. The application program 45 (Fig. 2) contains the constructs for forming the link labels in the browser running on the remote user station, and can vary depending upon the software content provided in the application. Hence, a programmer is free to design a web page display which can contain programmatic constructs such as if/then control statements. In this manner, the operations of application 45 are dependent upon the programmer's implementation of the software contained therein. In a JAVA platform, the application 45 could be implemented as an "applet" for downloading into a JAVA-enabled browser.

In accordance with the preferred embodiment of the present invention, the application program or module is adapted to operate in the browser program and contains a link label programmatically associated with a first network address. The application program can associate the link label with a second network address in response to user input at the remote user station.

Once the application is running in the browser, operation 64 displays a window, within the browser of the remote user station, having a link label (for example, 71, 72, and 73 in Figs. 3A-3C).

Because the link label of the present invention is defined within the application program downloaded from the central location, the link label is a programmatic construct which is characterized by a dynamic nature, as described above.

Input processing module 66 (Fig. 4) obtains user input with respect to a single link label displayed by operation 64 in the browser window. This input can include selection by the user of the link label, or more generally user activity within the browser window.

Link label modification module 68 alters the link label as dictated by the application program 45 in response to user input. Under the preferred embodiment of the present invention, the URL or text of the link label in the browser window is modified based on user activity within the browser window. Input processing module 66 and link label modification module 68 are also shown in Figs. 6, and 7A-7B.

Operation 70 then accesses the network with the appropriate data formed in the application. As will be described below, operation 70 can be programmed to

access the network after the user has armed and fired a link label which would require a transfer of data external to the application program running on the browser of the remote user station.

Figs. 5-7 illustrate the logical operations to implement the link label of the present invention. Fig. 5 illustrates the operations contained in application program 45 for creation and use of a link label. Operation 90 creates a new window within the browser for the display of information including the link label. Operation 92 creates a data display component associated with a link label, such as text field 75 in Fig. 3A, and operation 94 positions the data display component within the window.

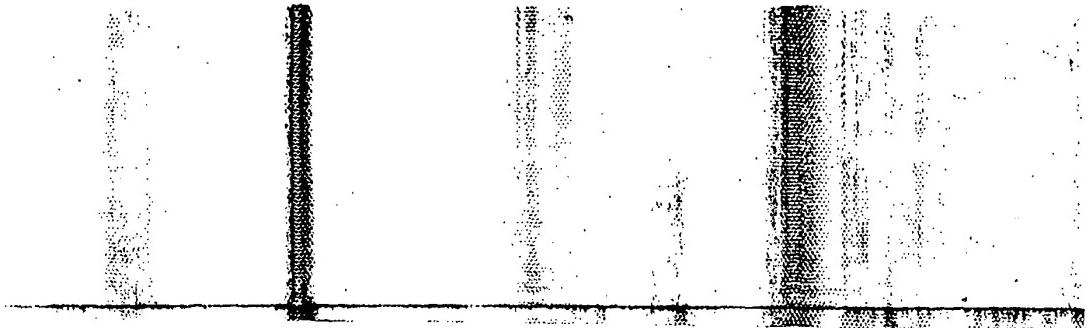
Operation 96 creates a link label, as a programmatic component of application program 45 (Fig. 2). Operation 98 positions the link label adjacent to the data display component, while operation 100 sets the URL and data parameters for the link label for subsequent use. Together these operations are used to place the link label within the window of a browser running on a remote user station.

Fig. 6 illustrates the logical operations of obtaining user input within a browser page and appropriately modifying the link label. These operations can also be included as part of application program 45 which is downloaded to the remote user station. Operation 110 obtains user input of data within the current browser window. The user input data can take the form of textual data, data from an input device such as a keyboard, mouse, or other forms of user input data. Decision operation 112 determines if the URL associated with the label should be changed responsive to the user input. If so, operation 114 alters or updates the URL associated with the link label to be loaded.

Operation 116 determines if the text of the link label should be changed. If so, operation 118 appropriately updates the text of the label. Operation 120 determines if the user data contains parameter data which can be passed to another application 45. If the user input data contains passable parameter data, then operation 122 formats the user data for subsequent processing. Decision operation 124 determines if the link label should be armed or loaded into the browser for communications with the device specified by the URL address contained in the link label. If so, operation 126 accesses the network with the correct URL information.

Figs. 7A-7B illustrate the logical operations of processing the user input into the browser window and appropriately utilizing the link label therefrom.

In Fig. 7A, operation 130 displays a browser window containing a link label. Operation 132 determines whether the link label is an active link label. Because of the programmatic nature of the link labels of the present invention, one feature of a link label is that it can be deactivated programmatically so that user selection of a deactivated link label would have no effect. If decision operation 132 determines that the link label is deactivated, then no further action is taken. Otherwise, the operations can proceed to operation 134.



11

EP 0 810 520 A1

12

an application module for creating a link label associated with a first network address, said application module adapted to associate said link label with a second network address in response to user input at the remote user station, said application module adapted to operate within said browser program;

an input processing module for processing user input received in the browser (50) during operation of the application module, said input processing module adapted to detect a relative location of an input device and the state of input buttons; and

an link label modification module for modifying the link label in response to said user input (51) at the remote user station (35) by associating said link label with a second network address.

20

25

30

35

40

45

50

55

EP 0 810 520 A1

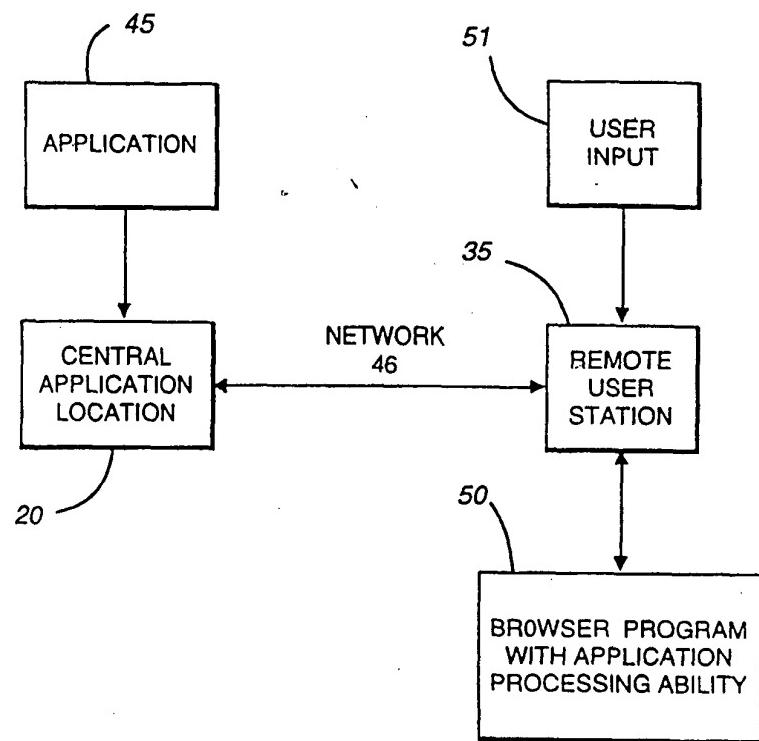


Fig. 2

EP 0 810 520 A1

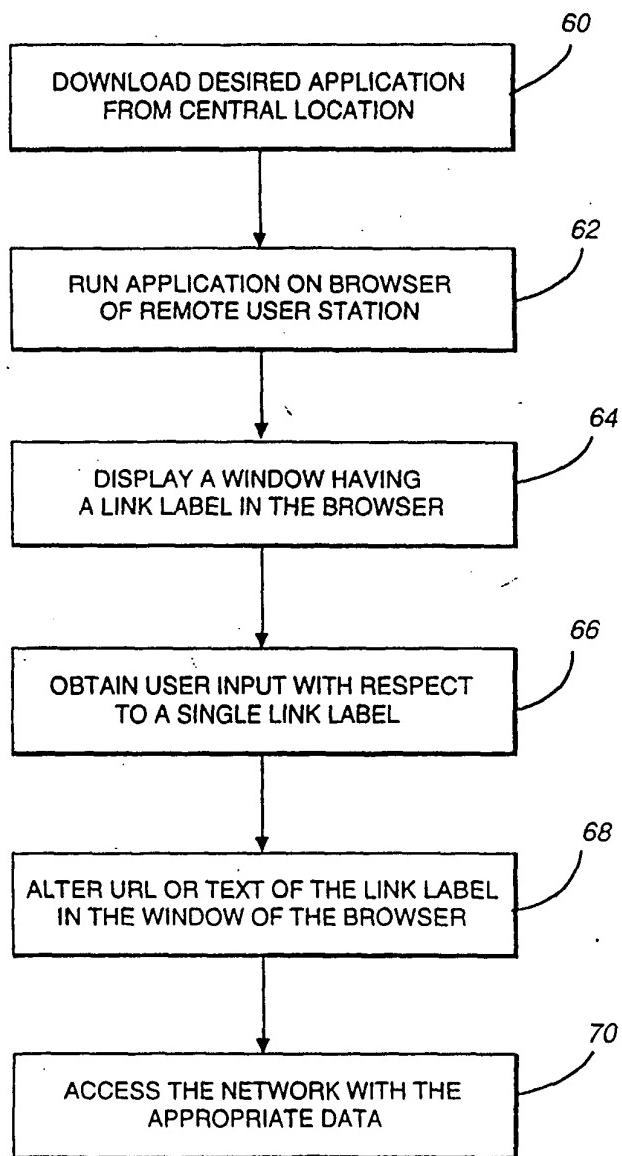


Fig. 4

EP 0 810 520 A1

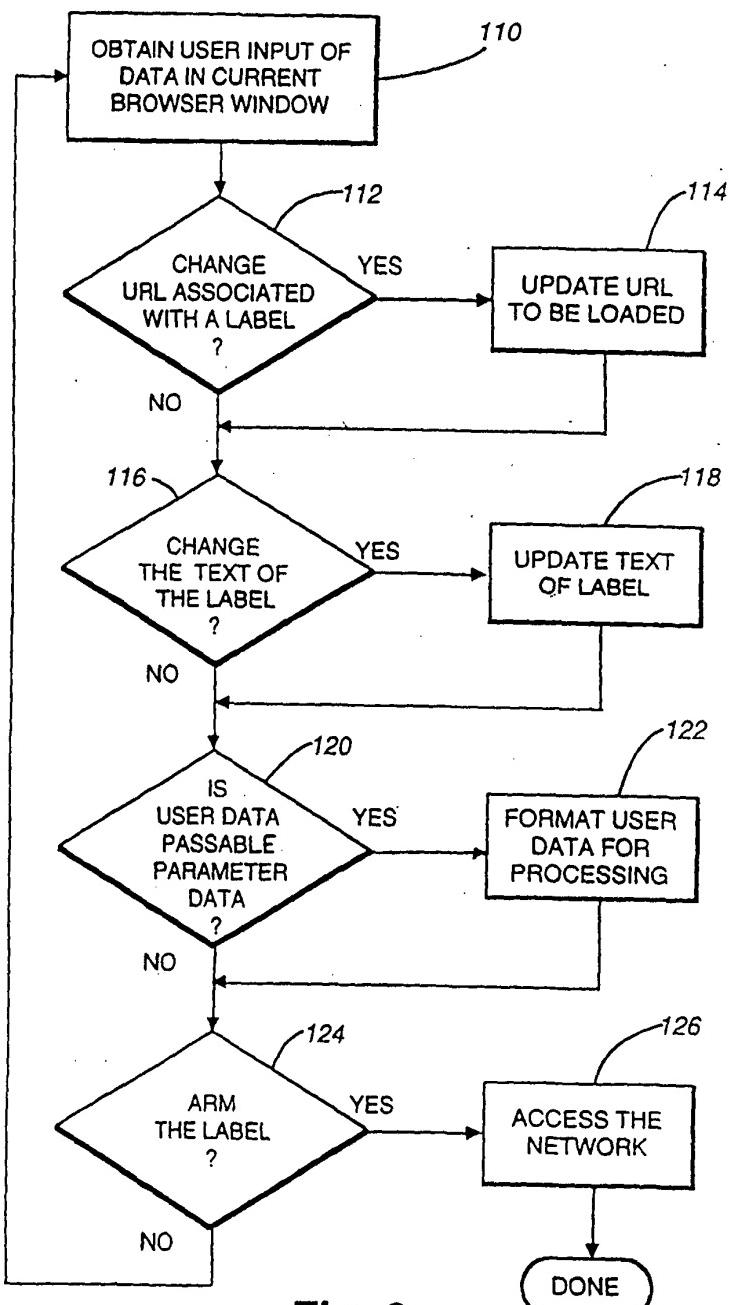


Fig. 6

EP 0 810 520 A1

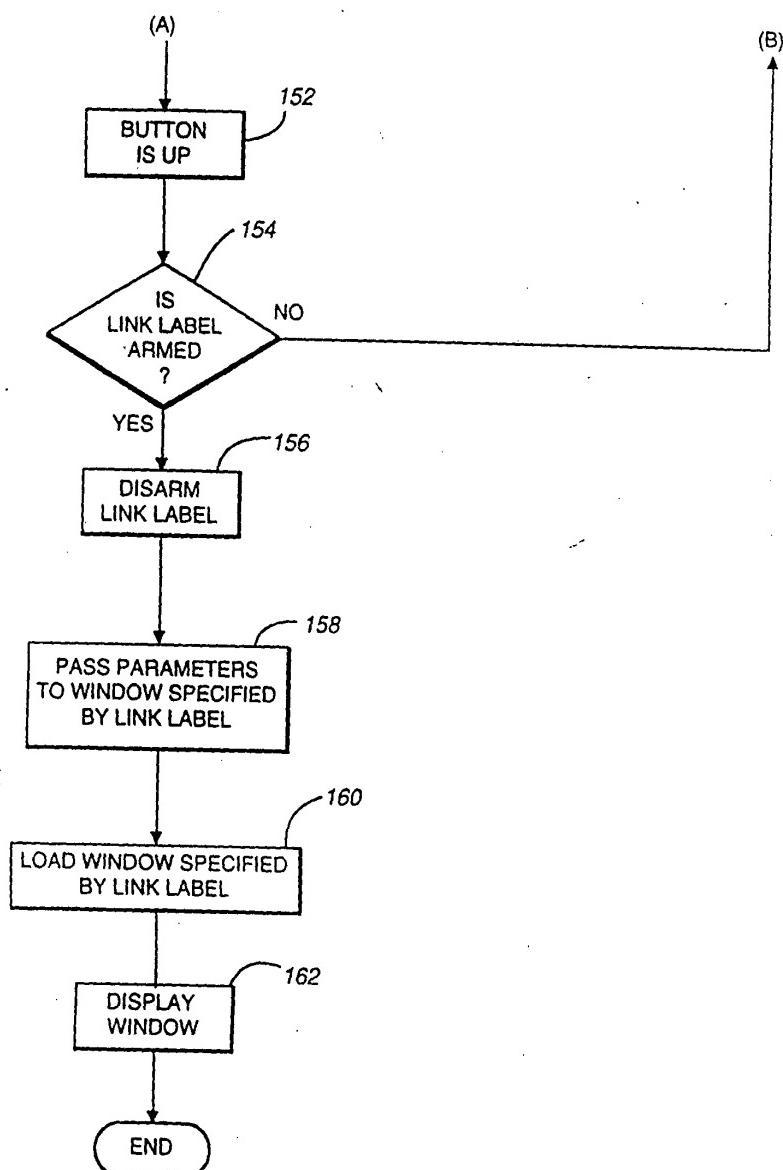


Fig. 7B

EP 0 810 520 A1



European Patent
Office

EUROPEAN SEARCH REPORT

Application Number
EP 97 20 1341

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.)
A	PROCEEDINGS, WORKSHOP ON MOBILE COMPUTING SYSTEMS AND APPLICATIONS, 8 December 1994, pages 185-190, XP000602490 VOELKER G M ET AL: "MOBISAIC: AN INFORMATION SYSTEM FOR A MOBILE WIRELESS COMPUTING ENVIRONMENT" * page 185, right-hand column, line 8 - line 30 * * page 186, right-hand column, line 24 - line 44 * * page 189, right-hand column, line 1 - line 6 * * page 187, paragraph 3.1 * ---	1,2	G06F9/44 G06F17/30
A	IEEE NETWORK: THE MAGAZINE OF COMPUTER COMMUNICATIONS, vol. 10, no. 2, 1 March 1996, pages 10-17, XP000580078 SCHULZRINNE H: "WORLD WIDE WEB: WHENCE, WHITHER, WHAT NEXT?" * page 15, right-hand column, line 26 - line 43 * * page 14, left-hand column, line 1 - line 27 * ---	1,2	TECHNICAL FIELDS SEARCHED (Int.Cl.) G06F
A	LIBRARY SOFTWARE REVIEW, vol. 13, no. 4, 1 January 1994, pages 269-279, XP000567351 DUVAL B K ET AL: "EXPLORING THE INTERNET WITH MOSAIC" * page 272, right-hand column, line 36 - page 273, left-hand column, line 2 * --- ---	1,2 -/-	
The present search report has been drawn up for all claims			
Place of search	Date of completion of the search	Examiner	
THE HAGUE	10 September 1997	Michel, T	
CATEGORY OF CITED DOCUMENTS		T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons A : member of the same patent family, corresponding document	
X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document			